

Compiladores

Aula 8

Análise Léxica

Flex – Gerador de Analisador Léxico

Prof. Dr. Luiz Eduardo G. Martins

UNIFESP



Flex – Gerador de Analisador Léxico

- O processo de construção de um analisador léxico pode ser automatizado
- Existem vários geradores de analisadores léxicos disponíveis gratuitamente, que ajudam muito no desenvolvimento de um compilador
- Um gerador muito conhecido é o *Flex (Fast Lex)*
 - Distribuído como parte do pacote de compilação *GNU* (produzido pela *Free Software Foundation*)

Flex – Gerador de Analisador Léxico

- Convenções de metacaracteres em *Lex*

Tabela 2.2 Convenções de metacaracteres em *Lex*

Padrão	Significado
<i>a</i>	caractere <i>a</i>
" <i>a</i> "	caractere <i>a</i> , mesmo se <i>a</i> for um metacaractere
\ <i>a</i>	caractere <i>a</i> se <i>a</i> for um metacaractere (pela interpretação ANSI –C)
<i>a</i> *	zero ou mais repetições de <i>a</i>
<i>a</i> +	uma ou mais repetições de <i>a</i>
<i>a</i> ?	um <i>a</i> opcional
<i>a</i> <i>b</i>	<i>a</i> ou <i>b</i>
(<i>a</i>)	<i>a</i> propriamente dito
[<i>abc</i>]	qualquer caractere entre <i>a</i> , <i>b</i> e <i>c</i>
[<i>a–d</i>]	qualquer caractere entre <i>a</i> , <i>b</i> , <i>c</i> e <i>d</i>
[^ <i>ab</i>]	qualquer caractere, exceto <i>a</i> ou <i>b</i>
.	qualquer caractere, exceto mudança de linha
{ <i>xxx</i> }	a expressão regular representada pelo nome <i>xxx</i>

Flex – Gerador de Analisador Léxico

- Formato do arquivo de entrada *Flex*
 - O arquivo de entrada é composto por três partes
 - Definições
 - Regras
 - Rotinas do usuário
 - As seções são separadas por dois sinais de porcentagem, que aparecem em linhas separadas, iniciando na primeira coluna

Flex – Gerador de Analisador Léxico

- Formato do arquivo de entrada *Lex*

%{

qualquer código C a ser inserido externamente a

qualquer função opcional

%}

definições regulares opcional

%%

expressões regulares seguidas do código C a ser executado quando houver
casamento com a expressão regular correspondente

%%

código C para rotinas auxiliares ativadas pela segunda seção, pode também
conter um programa principal opcional

Flex – Gerador de Analisador Léxico

- Nomes internos utilizados por *Flex*

Tabela 2.3 Alguns nomes internos Lex

Nome Interno Lex	Significado/Utilização
<code>lex.yy.c</code> or <code>lexyy.c</code>	Arquivo de saída Lex
<code>yylex</code>	Rotina de varredura Lex
<code>yytext</code>	Cadeia casou com ação corrente
<code>yyin</code>	Entrada Lex (padrão: <code>stdin</code>)
<code>yyout</code>	Saída Lex (padrão: <code>stdout</code>)
<code>input</code>	Rotina de entrada com reservatório Lex
<code>ECHO</code>	Ação básica Lex (imprime <code>yytext</code> em <code>yyout</code>)

- A documentação completa de *FLEX* está disponível em <http://flex.sourceforge.net/manual/>

Flex – Gerador de Analisador Léxico

- Exemplo de utilização do Flex (arquivo de entrada para o Flex)

```
/*
 * Description: Count the number of characters and the number of lines
 *             from standard input
 * */
%{
int num_lines = 0, num_chars = 0;
}%

%%
\n  ++num_lines; ++num_chars;
.   ++num_chars;
fim return 0;
%%

main()
{
  yylex();
  printf("# of lines = %d, # of chars = %d\n", num_lines, num_chars);
}
```

Flex – Gerador de Analisador Léxico

- Exemplo de utilização do *Flex*
 - Reconhecimento de cadeias que atendam a especificação de números inteiros nos formatos
 - Decimal
 - Octal
 - Hexadecimal
 - Binário

Flex – Gerador de Analisador Léxico

- Exemplo de utilização do *Flex*

```
DIGIT [0-9]
%%
[1-9]{DIGIT}*
0[0-7]*
0x[0-9A-Fa-f]+
0b[01]+
<<EOF>>
%%
int main(int argc, char *argv[])
{
    FILE *f_in;

    if (argc == 2)
    {
        if(f_in = fopen(argv[1], "r")) yyin = f_in;
        else perror(argv[0]);
    }
    else yyin = stdin;

    yylex();
    return(0);
}
```

```
printf("DEC");
printf("OCT");
printf("HEX");
printf("BIN");
return 0;
```

- Arquivo de entrada (programa), a ser lido pelo gerador de analisador léxico (*Flex*)
- Este arquivo deve ter extensão .l
- No exemplo, o nome do arquivo é *exemplo.l*

Flex – Gerador de Analisador Léxico

- Exemplo de utilização do *Flex*
 - Ao digitar:
~\$ flex exemplo.l
 - Flex gerará o analisador léxico, com o seguinte nome padrão
lex.yy.c
 - *lex.yy.c* implementa os procedimentos do analisador léxico, mas ainda precisa ser compilado e ligado com a biblioteca *libfl*
~\$ gcc -o exec lex.yy.c -lfl

Flex – Gerador de Analisador Léxico

- Exemplo de utilização do *Flex*
 - Podemos executar *exec* com ou sem um arquivo texto de entrada
 - Sem arquivo de entrada
 - exec* fará a leitura de caracteres de entrada via **teclado**
(*yyin* = *stdin*)
 - Com arquivo de entrada
 - exec* fará a leitura de caracteres de entrada via **arquivo**
(*yyin* = *f_in*)

Flex – Gerador de Analisador Léxico

- Exemplo de utilização do *Flex*
 - Considere um arquivo de entrada que contenha as seguintes cadeias de caracteres:

12345

02343

0xFA

0b11

rere



exec



DEC

OCT

HEX

BIN

rere

tokens
válidos

Flex – Gerador de Analisador Léxico

- Bibliografia consultada

LOUDEN, K. C. **Compiladores: princípios e práticas.**

São Paulo: Pioneira Thompson Learning, 2004

RICARTE, I. **Introdução à Compilação.** Rio de Janeiro:
Editora Campus/Elsevier, 2008.